UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/627,518	07/28/2000	Gennaro A. Cuomo	RSW9-2000-0090-US1	3351

7590 01/30/2004

A Bruce Clay
IBM Corporation Dept T81 062
3039 Cornwallis Road
Research Triangle Park, NC 27709

EXAMINER	
MAURO JR, THOMAS J	
ART UNIT	**PAPER NUMBER**
2143	6

DATE MAILED: 01/30/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/627,518	CUOMO ET AL.	
	Examiner	Art Unit	
	Thomas J. Mauro Jr.	2143	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *15 December 2003*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-26* is/are pending in the application.

4a) Of the above claim(s) *6,16 and 21-24* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-5,7-15,17-20,25 and 26* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. §§ 119 and 120

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

13)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

a) ☐ The translation of the foreign language provisional application has been received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449) Paper No(s) *2*.
4) ☒ Interview Summary (PTO-413) Paper No(s). *7*.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: .

DETAILED ACTION

1. This action is responsive to the amendment of the application (Paper No. 5) filed on 12/15/03. Claims 1-26 are presented for further examination.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1, 2, 5 and 7 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008) and Johnson et al. (U.S. 6,591,250).

Regarding claim 1, Chung teaches the invention substantially as claimed, a method in a data processing system for managing a request including a session identification, comprising:

calculating a first value based on the session identification **[Chung -- Col. 4 lines 37-39 – The client IP address which acts as the session identifier has a first value calculated from it by performing a hash function on the session identifier]**;

routing the request to a first server based on the first value **[Chung -- Col. 4 lines 59-63 – The computed hash value from the session identifier is compared to a list of servers to determine which server should handle the request]**;

Chung fails to teach determining whether the first server is functional.

Khuc, however, teaches a routing system which determines whether the first server is functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include the teachings of Khuc for determining whether a server is functional or not and routing the request based on that information into the invention of Chung et al. in order to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

In addition, Chung teaches calculating a second value in response to the first server being non-functional and routing the request to a second server **[Chung -- Col. 7 lines 9-12 – When a server is down, the dispatch will rehash the IP address, i.e. session identifier, and route this and all subsequent packets to the newly mapped server to prevent any lost data packets caused by the failure]**. Chung, however, fails to teach that the second value is calculated based on the first value.

Johnson, however, teaches calculating a second value based upon a first value which entails performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented.

It would have been obvious to one of ordinary skill in the art at the time the invention was made

to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

Regarding claim 2, Chung-Khuc-Johnson teach the invention substantially as claimed, as aforementioned in claim 1 above, wherein the step of calculating a first value comprises performing a hash function on the session identification **[Chung -- Col. 4 lines 37-39 – An appropriate hash function is used on the client IP address, also known as, the session identifier]**.

Regarding claim 5, Chung-Khuc-Johnson teach the invention substantially as claimed, as aforementioned in claim 1 above, wherein the step of routing the request to a server comprises:

selecting a first server based on the first value **[Chung -- Col. 4 lines 59-63 – The computed hash value (first value) from the session identifier is compared to a list of servers to determine which server should handle the request]**;

determining whether the first server is functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**; and

routing the request to the first server **[Chung -- Col. 4 lines 59-63 – The computed hash value from the session identifier is compared to a list of servers to determine which server**

should handle the request] in response to the first server being functional **[Khuc -- Col. 9 lines 32-34 – As long as the percent allocation, as described above, is not "0", the routing systems will route the request to the primary server because it is functional]**.

Regarding claim 7, Chung-Khuc-Johnson teach the invention substantially as claimed, as aforementioned in claim 5 above, including determining whether the first server is functional using a look-up table **[Khuc -- Figure 5 & Col. 9 lines 37-39 – By comparing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

4. Claims 3 and 4 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389), Khuc (U.S. 6,470,008) and Johnson et al. (U.S. 6,591,250), as applied to claims 2 and 3 above, respectively, in view of Muller et al. (U.S. 6,606,301).

Regarding claim 3, Chung-Khuc-Johnson teaches the invention substantially as claimed, as aforementioned in claim 2 above, but fail to teach performing a modulus function to form an integer and selecting a server based on the integer.

Muller teaches performing a modulus function on the first value to form a first integer **[Muller -- Col. 49 lines 52-53 – After hashing, a modulus function is performed. This, by the function's nature, produces an integer value]**.

Chung performs a hash function on a session identifier, i.e. IP address, in order to obtain greater speed and efficiency to route requests to a plurality of servers and to prevent traffic and bottlenecks.

Muller, after hashing a value, further performs a modulus function to further provide more speed and efficiency in distributing the processing requests and to prevent "bottlenecks" **[Muller -- Col. 49 lines 52-54]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate performing a modulus function on a hash value, as taught by Muller into the routing system of Chung-Khuc-Johnson, in order to further provide increased speed and efficiency in routing requests and to prevent bottlenecks from occurring.

In addition, Chung fails to teach selecting a server based upon the integer.

Khuc teaches a look-up table which uses integers as the index into the table to determine which server to route requests to **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**.

Both Chung and Khuc are both interested in quickly and efficiently routing requests to servers. Khuc uses a look-up table to further speed up the routing process.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the routing of request using integers in a look-up table, as taught by Khuc into the invention of Chung-Khuc-Johnson-Muller in order to further provide a fast and efficient method for quickly finding and routing requests to the correct server.

Regarding claim 4, Chung-Khuc-Johnson-Muller teach the invention substantially as claimed, as aforementioned in claim 3 above, wherein the step of selecting a server comprises looking up the server in a look-up table using the first integer **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**.

5. Claims 8-10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008), Johnson et al. (U.S. 6,591,250) and Muller et al. (U.S. 6,606,301).

Regarding claim 8, Chung teaches the invention substantially as claimed, a method in a data processing system for routing a request to one of a number of servers, comprising:

receiving a request including a session identification **[Chung -- Col. 7 lines 64-66 – The request packet contains the client IP address (session identification]**;

performing a hash function on the session identification to form a first hash value **[Chung -- Col. 4 lines 37-39 and Col. 7 lines 64-66 – An appropriate hash function is used on the client IP address, also known as, the session identifier]**.

Chung fails to teach performing a modulus function on a first and second hash value to form a first and second integer.

Muller teaches performing a modulus function on the first or second value to form a first/second

integer **[Muller -- Col. 49 lines 52-53 – After hashing, a modulus function is performed. This, by the function's nature, produces an integer value]**.

Chung performs a hash function on a session identifier, i.e. IP address, in order to obtain greater speed and efficiency to route requests to a plurality of servers and to prevent traffic and bottlenecks.

Muller, after hashing a value, further performs a modulus function to further provide more speed and efficiency in distributing the processing requests and to prevent "bottlenecks" **[Muller -- Col. 49 lines 52-54]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate performing a modulus function on a hash value, as taught by Muller into the routing system of Chung, in order to further provide increased speed and efficiency in routing requests and to prevent bottlenecks from occurring.

In addition, Chung fails to teach routing the request to a first/second server using the first/second integer in response to the first server being functional.

Khuc teaches a look-up table which uses integers as the index into the table to determine which server to route requests to **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**, in response to a first server being functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

Both Chung and Khuc are both interested in quickly and efficiently routing requests to servers. Khuc uses a look-up table to further speed up the routing process.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the routing of request using integers in a look-up table after determining if the server is functional, as taught by Khuc into the invention of Chung in order to further provide a fast and efficient method for quickly finding and routing requests to the correct server and to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

Furthermore, Chung fails to teach performing a hash function on the first hash value to form a second hash value.

Johnson, however, teaches calculating a second value based upon a first value which entails performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

Regarding claim 9, Chung-Khuc-Johnson-Muller teach the invention substantially as claimed, as aforementioned in claim 8 above, wherein the integer is between zero and the number of servers minus one **[Muller -- Col. 49 lines 52-54 and lines 63-66 – By performing the modulus function over the hash function result using the number of available components, i.e. servers, the resulting integer must be between 0 and N-1 components, i.e. servers]**.

Regarding claim 10, Chung-Khuc-Johnson-Muller teach the invention substantially as claimed, as aforementioned in claim 8 above, wherein the step of routing the request comprises looking up the server in a look-up table using the integer **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that the client should be routed to in order to service the request]**.

6. Claims 11, 12, 15 and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008) and Johnson et al. (U.S. 6,591,250).

Regarding claim 11, Chung teaches the invention substantially as claimed, an apparatus in a data processing system for managing a request including a session identification, comprising:

calculating a first value based on the session identification **[Chung -- Col. 4 lines 37-39 – The client IP address which acts as the session identifier has a first value calculated from it by performing a hash function on the session identifier]**;

routing the request to a first server based on the first value **[Chung -- Col. 4 lines 59-63 – The computed hash value from the session identifier is compared to a list of servers to determine which server should handle the request]**;

Chung fails to teach determining whether the first server is functional.

Khuc, however, teaches a routing system which determines whether the first server is functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include the teachings of Khuc for determining whether a server is functional or not and routing the request based on that information into the invention of Chung et al. in order to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

In addition, Chung teaches calculating a second value in response to the first server being non-functional and routing the request to a second server **[Chung -- Col. 7 lines 9-12 – When a server is down, the dispatch will rehash the IP address, i.e. session identifier, and route this and all subsequent packets to the newly mapped server to prevent any lost data packets caused by the failure]**. Chung, however, fails to teach that the second value is calculated based on the first value.

Johnson, however, teaches calculating a second value based upon a first value which entails

performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

Regarding claims 12 and 15, these are apparatus claims corresponding to the methods claimed in claims 2 and 5. They have similar limitations; therefore, claims 12 and 15 are rejected under the same rationale.

Regarding claim 17, this is an apparatus claims corresponding to the method claimed in claim 7. It has similar limitations; therefore, claim 17 is rejected under the same rationale.

7. Claims 13 and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389), Khuc (U.S. 6,470,008) and Johnson et al. (U.S. 6,591,250), as applied to claims 12 and 13 above, respectively, in view of Muller et al. (U.S. 6,606,301).

Regarding claim 13, Chung-Khuc-Johnson teaches the invention substantially as claimed, as aforementioned in claim 12 above, but fail to teach performing a modulus function to form an integer and selecting a server based on the integer.

Muller teaches performing a modulus function on the first value to form a first integer **[Muller -- Col. 49 lines 52-53 – After hashing, a modulus function is performed. This, by the function's nature, produces an integer value]**.

Chung performs a hash function on a session identifier, i.e. IP address, in order to obtain greater speed and efficiency to route requests to a plurality of servers and to prevent traffic and bottlenecks.

Muller, after hashing a value, further performs a modulus function to further provide more speed and efficiency in distributing the processing requests and to prevent "bottlenecks" **[Muller -- Col. 49 lines 52-54]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate performing a modulus function on a hash value, as taught by Muller into the routing system of Chung-Khuc-Johnson, in order to further provide increased speed and efficiency in routing requests and to prevent bottlenecks from occurring.

In addition, Chung fails to teach selecting a server based upon the integer.

Khuc teaches a look-up table which uses integers as the index into the table to determine which server to route requests to **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**.

Both Chung and Khuc are both interested in quickly and efficiently routing requests to servers. Khuc uses a look-up table to further speed up the routing process.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the routing of request using integers in a look-up table, as taught by Khuc into the invention of Chung-Khuc-Johnson-Muller in order to further provide a fast and efficient method for quickly finding and routing requests to the correct server.

Regarding claim 14, this is an apparatus claims corresponding to the method claimed in claim 4. It has similar limitations; therefore, claim 14 is rejected under the same rationale.

8. Claims 18-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008), Johnson et al. (U.S. 6,591,250) and Muller et al. (U.S. 6,606,301).

Regarding claim 18, Chung teaches the invention substantially as claimed, an apparatus in a data processing system for routing a request to one of a number of servers, comprising:

a processor; and

a memory electrically connected to the processor, the memory having stored therein a program to be executed on the processor **[Chung -- Col. 7 lines 45-47 – By having an operating system (OS), it is inherent that in order for the OS to process tasks and store/run**

not only itself but also the routing algorithm correctly, a processor and memory must be contained within the dispatcher];

receiving a request including a session identification **[Chung -- Col. 7 lines 64-66 – The request packet contains the client IP address (session identification]**;

performing a hash function on the session identification to form a first hash value **[Chung -- Col. 4 lines 37-39 and Col. 7 lines 64-66 – An appropriate hash function is used on the client IP address, also known as, the session identifier]**.

Chung fails to teach performing a modulus function on a first and second hash value to form a first and second integer.

Muller teaches performing a modulus function on the first or second value to form a first/second integer **[Muller -- Col. 49 lines 52-53 – After hashing, a modulus function is performed. This, by the function's nature, produces an integer value]**.

Chung performs a hash function on a session identifier, i.e. IP address, in order to obtain greater speed and efficiency to route requests to a plurality of servers and to prevent traffic and bottlenecks.

Muller, after hashing a value, further performs a modulus function to further provide more speed and efficiency in distributing the processing requests and to prevent "bottlenecks" **[Muller -- Col. 49 lines 52-54]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate performing a modulus function on a hash value, as taught by Muller into the routing system of Chung, in order to further provide increased speed and efficiency in routing requests and to prevent bottlenecks from occurring.

In addition, Chung fails to teach routing the request to a first/second server using the first/second integer in response to the first server being functional.

Khuc teaches a look-up table which uses integers as the index into the table to determine which server to route requests to **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**, in response to a first server being functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

Both Chung and Khuc are both interested in quickly and efficiently routing requests to servers. Khuc uses a look-up table to further speed up the routing process.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the routing of request using integers in a look-up table after determining if the server is functional, as taught by Khuc into the invention of Chung in order to further provide a fast and efficient method for quickly finding and routing requests to the correct server and to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

Furthermore, Chung fails to teach performing a hash function on the first hash value to form a second hash value.

Johnson, however, teaches calculating a second value based upon a first value which entails performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

Regarding claims 19 and 20, these are apparatus claims corresponding to the methods claimed in claims 9 and 10. They have similar limitations; therefore, claims 19 and 20 are rejected under the same rationale.

9. Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008) and Johnson et al. (U.S. 6,591,250).

Regarding claim 25, Chung teaches the invention substantially as claimed, a computer program product, in a computer readable medium, for managing a request including a session identification **[Chung -- Col. 7 lines 45-47 - A dispatcher which includes an operating system, i.e. server which inherently contains instructions for carrying out the dispatching tasks]**, comprising:

calculating a first value based on the session identification **[Chung -- Col. 4 lines 37-39 – The client IP address which acts as the session identifier has a first value calculated from it by performing a hash function on the session identifier]**;

routing the request to a first server based on the first value **[Chung -- Col. 4 lines 59-63 – The computed hash value from the session identifier is compared to a list of servers to determine which server should handle the request]**;

Chung fails to teach determining whether the first server is functional.

Khuc, however, teaches a routing system which determines whether the first server is functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include the teachings of Khuc for determining whether a server is functional or not and routing the request based on that information into the invention of Chung et al. in order to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

In addition, Chung teaches calculating a second value in response to the first server being non-functional and routing the request to a second server **[Chung -- Col. 7 lines 9-12 – When a server is down, the dispatch will rehash the IP address, i.e. session identifier, and route this and all subsequent packets to the newly mapped server to prevent any lost data packets caused by the failure]**. Chung, however, fails to teach that the second value is calculated based on the first value.

Johnson, however, teaches calculating a second value based upon a first value which entails

performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

10. Claim 26 is rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008), Johnson et al. (U.S. 6,591,250) and Muller et al. (U.S. 6,606,301).

Regarding claim 26, Chung teaches the invention substantially as claimed, a computer program product, in a computer readable medium, for managing a request including a session identification **[Chung -- Col. 7 lines 45-47 - A dispatcher which includes an operating system, i.e. server which inherently contains instructions for carrying out the dispatching tasks]**, comprising:

receiving a request including a session identification **[Chung -- Col. 7 lines 64-66 – The request packet contains the client IP address (session identification]**;

performing a hash function on the session identification to form a first hash value **[Chung -- Col. 4 lines 37-39 and Col. 7 lines 64-66 – An appropriate hash function is used on the client IP address, also known as, the session identifier]**.

Chung fails to teach performing a modulus function on a first and second hash value to form a first and second integer.

Muller teaches performing a modulus function on the first or second value to form a first/second integer **[Muller -- Col. 49 lines 52-53 – After hashing, a modulus function is performed. This, by the function's nature, produces an integer value]**.

Chung performs a hash function on a session identifier, i.e. IP address, in order to obtain greater speed and efficiency to route requests to a plurality of servers and to prevent traffic and bottlenecks.

Muller, after hashing a value, further performs a modulus function to further provide more speed and efficiency in distributing the processing requests and to prevent "bottlenecks" **[Muller -- Col. 49 lines 52-54]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate performing a modulus function on a hash value, as taught by Muller into the routing system of Chung, in order to further provide increased speed and efficiency in routing requests and to prevent bottlenecks from occurring.

In addition, Chung fails to teach routing the request to a first/second server using the first/second integer in response to the first server being functional.

Khuc teaches a look-up table which uses integers as the index into the table to determine which server to route requests to **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**, in response to a first server being functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

Both Chung and Khuc are both interested in quickly and efficiently routing requests to servers. Khuc uses a look-up table to further speed up the routing process.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the routing of request using integers in a look-up table after determining if the server is functional, as taught by Khuc into the invention of Chung in order to further provide a fast and efficient method for quickly finding and routing requests to the correct server and to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

Furthermore, Chung fails to teach performing a hash function on the first hash value to form a second hash value.

Johnson, however, teaches calculating a second value based upon a first value which entails performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented. It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

Response to Arguments

11. Applicant's arguments filed 12/15/2003 have been fully considered but they are not persuasive.

(A) Chung teaches creating a hash value from the IP address of the client in order to route a request to a given server. Johnson teaches taking the hashed value and re-hashing the value to produce a new hash value. Applicant claims that this combination of Chung and Johnson would not be obvious to one skilled in the art.

In response to argument A, the combination of Chung and Johnson would be obvious to a person of ordinary skill in the art at the time the invention was made. Chung, as a whole, discloses a method of routing requests through a dispatcher to various servers. Server side components in his system include a router, dispatcher, and the various servers comprising a server farm **[Chung -- Figure 4]**. Chung teaches that in the event that a server is non-functional,

i.e. a server has failed, a second hash is calculated in order to route the request to a different server **[Chung -- Col. 7 lines 9-12]**. As stated in the above rejection, upon which Examiner draws the applicant's attention, Johnson teaches the deficiency in the Chung reference that a second hash value is calculated from the first hash value **[Johnson -- Col. 13 lines 30-32]**. While there can be various reasons why a server has failed, one all too common reason stems from an outside attack or virus that has occurred, which has left one or more servers non-functional. Therefore, Chung would see the need for added security on the server side components of his invention in order to prevent such attacks from occurring. Johnson teaches that a second hash value is calculated from the first hash value in order to increase the security with which information is transmitted or stored. With this knowledge, Chung would see that by calculating a second hash on the already hashed value would provide further security in that the new server upon which it was being routed (a functional server) would be further disguised. This would provide added protection and prevent any other servers from becoming non-functional in the event of another staged attack. Therefore, the Examiner accordingly demurs to this assertion because the combination would have been obvious to one of ordinary skill in the art for the reasons stated above and in the office action.

(B) Chung teaches that a second hash value is created using an altered hash function on the IP address, whereas claim 1 calls for calculating a second value based on the first value. Applicant argues that Chung uses a different hash function whereas applicant uses the "same hash function."

In response to argument B, while applicant argues that "the same hash function" is used to calculate the second value, nowhere does this statement appear in the claims. As is stated in claim 1, "a second hash value is calculated from the first value." Therefore, the applicant's argument with regards to "the same hash function" does not commensurate with the claim language.

(C) Chung teaches that a hash value is calculated from an IP address which is used to dispatch to a given server. Muller teaches that a modulus function is used to route a given request to one of a number of processors. Applicant argues that the combination of Chung and Muller would not be obvious to one skilled in the art.

In response to argument C, the combination of Chung and Muller would be obvious to a person of ordinary skill in the art at the time the invention was made. Chung, as a whole, discloses a method of routing requests through a dispatcher to various servers. In order to route the requests to the various servers with speed and efficiency, Chung hashes the IP address of the client and uses the hashed value to select a server. By hashing the value, a smaller, whole number is obtained which can quickly and efficiently be used to select a server and route a request. Thus, Chung wishes to obtain speed and efficiency in routing, which he does obtain by using a hash function. Similarly, Muller discloses using a modulus function to after he obtains a hash value to further obtain a smaller value with which to route requests to one of a number of processors, i.e. servers. Muller uses the modulus function in order to obtain a smaller value

which he further discloses provides for increased speed in routing to the various number of processors. Therefore, because both Chung and Muller are calculating values to route to different processors or servers and because both are doing it to obtain speed and efficiency and to prevent "bottleneck" situations from developing, it would have been obvious to combine the two references. Chung would want to incorporate the modulus function in order to further gain speed and efficiency in addition to doing his hash function. Therefore, the Examiner accordingly demurs to this assertion because the combination would have been obvious to one of ordinary skill in the art for the reasons stated above and in the office action.

(D) Chung teaches that a hash value is calculated from an IP address which is used to dispatch or route the request to a given server. Khuc teaches the use of a look-up table to use in routing requests to various servers. Applicant argues that the combination of Chung and Khuc would not be obvious to one skilled in the art.

In response to argument D, the combination of Chung and Khuc would be obvious to a person of ordinary skill in the art at the time the invention was made. Chung, as a whole, discloses a method of routing requests through a dispatcher to various servers. In order to route the requests to the various servers with speed and efficiency, Chung hashes the IP address of the client and uses the hashed value to select a server. By hashing the value, a smaller, whole number is obtained which can quickly and efficiently be used to select a server and route a request. Thus, Chung wishes to obtain speed and efficiency in routing, which he does obtain by

using a hash function. Khuc, as is well known in the art, teaches the use of a look-up table to quickly and efficiently route requests. Because this technique of using a routing look-up table was well known in the art and Chung wants to gain speed and efficiency for routing, something a look-up table provides, it would have been obvious to one of ordinary skill in the art for Chung to incorporate the look-up table teachings of Khuc. Therefore, the Examiner accordingly demurs to this assertion because the combination would have been obvious to one of ordinary skill in the art for the reasons stated above and in the office action.

Conclusion

12. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Thomas J. Mauro Jr. whose telephone number is 703-605-1234. The examiner can normally be reached on M-F 8:00a.m. - 4:30p.m..

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, David A. Wiley can be reached on 703-308-5221. The fax phone number for the organization where this application or proceeding is assigned is 703-746-7239.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 703-305-3900.

TJM
January 20, 2004

DAVID WILEY
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100